Effect of Defuzzification Methods in Redundancy Allocation Problem with Fuzzy Valued Reliabilities via Genetic Algorithm

Sanat Kumar Mahato*[1], Laxminarayan Sahoo[2], Asoke Kumar Bhunia[3]

Department of Mathematics, Durgapur Govt. College
Durgapur-713214, West Bengal, India

Department of Mathematics, Raniganj Girls' College
Raniganj-713358, West Bengal, India

Department of Mathematics, The University of Burdwan
Burdwan-713104, West Bengal, India

[1]sanatkmahato@gmail.com; [2]lxsahoo@gmail.com; [3]bhuniaak@rediffmail.com

Abstract

This paper deals with redundancy allocation problem of complex system with ten subsystems connected in parallel with identical components. The reliability of each component is considered to be imprecise. This impreciseness has been represented by different fuzzy numbers. Then the corresponding problem has been transformed to crisp constrained optimization problem using different defuzzification methods. The transformed problem has been converted into unconstrained optimization problem by Big-M penalty technique, then solved by real coded genetic algorithm with advanced operators. Finally, to illustrate the methodology, a numerical example has been applied for different defuzzification methods and the obtained results have been compared.

Keywords

Reliability-redundancy Allocation; Genetic Algorithm; Fuzzy Number; Defuzzification; Penalty Function

Introduction

Due to the development of modern technology, the design of a system is dependent on the selection of components and configurations to meet the requirement as well as the performance. For a system with known cost, reliability, weight, volume and other parameters, the corresponding design problem is known as combinatorial optimization problem. The well known reliability design problem is referred to as redundancy allocation problem whose fundamental objective is to determine the number of redundant components that maximize the system reliability or minimize the system cost subject to the several resource constraints. As redundancy allocation problem is a nonlinear integer programming problem, it can not be solved by direct/indirect or mixed search methods due to discrete search space. This redundancy allocation problem is NP-hard and has been well discussed by Tillman et al. (1977a) and Kuo and Prasad (2001). In the existing literature, it is observed that several methods, viz. heuristic methods (Nakagawa and Nakashima 1977; Kim and Yum 1993; Kuo et al. 1978; Aggarwal and Gupta 2005; Ha and Kuo 2006), branch and bound method (Kuo et al. 1987; Tillman et al. 1977; Sun and Li 2002 ; Sung and Cho 1999), reduced gradient method (Hwang et al. 1979), integer programming method (Misra and Sharma 1991), dynamic programming (Nakagawa et al. 1981; Hikita et al. 1992) etc. were used to solve such redundancy allocation problem. However, these methods have some advantages as well as disadvantages. Dynamic programming technique can not be used in solving complicated problem as it can not be decomposable according to 'Bellman's optimality principle'. In branch and bound method, the effectiveness depends on the sharpness of the bound and required memory increases exponentially with the problem size. As a result, with the development of genetic algorithm and other evolutionary algorithms, most of the researchers are motivated to use these methods in solving redundancy allocation problem. These methods provide more flexibility, require less assumptions on the objective as well as constraints and more efficient irrespective of whether the search space is discrete or not.

In almost all the studies mentioned earlier, the parameters of redundancy allocation problem have usually been taken to be precise values. This means that complete probabilistic information about the system is

known. In this situation, it is commonly assumed that the value of probability is perfectly determinable for every event. However, in reality it is not true. There are not sufficient data available in most of the cases when the system is either new or it exists only as a project. In this case the data/ information can not be collected precisely due to human errors, improper storage facilities and other unexpected factors related to environment. So the reliability of a component of a system will be an imprecise number. To take the problem with such imprecise numbers, generally stochastic, fuzzy and fuzzy-stochastic approaches are applied and the corresponding problems are converted to deterministic problems for solving them.

In this paper, redundancy allocation problem of complicated system with ten subsystems has taken into consideration. Each subsystem is connected parallelly with identical components. The reliability of each component is considered to be fuzzy number. Then the corresponding problem has been transformed to crisp problem using different defuzzification methods. The transformed problem has been converted into unconstrained optimization problem by Big-M penalty technique, then solved by real coded genetic algorithm for integer variables with tournament selection, intermediate crossover and one neighbourhood mutation. Finally, to illustrate the methodology, a numerical example has been listed for different defuzzification methods and the computed results have been compared.

Representation of Fuzzy Number

The word 'fuzzy' was first introduced by Zadeh in 1965 in his famous research paper "Fuzzy Sets" as a mathematical way to represent impreciseness/ fuzziness or vagueness. The approach of fuzzy set is an extension of classical set theory and used in fuzzy logic. In classical set theory, the membership of each element in relation to a set is assessed in binary terms according to a crisp condition; and an element either belongs to or not to the set. By contrast, a fuzzy set theory permits the gradual assessment of the membership of each element in relation to a set; which is discussed with the aid of a membership function. Fuzzy set is an extension of classical set theory since, for a certain universe, a membership function may act as an indicator function, mapping all elements to either 1 or 0, as in the classical notation. He used this word to generalize the mathematical concept of the set to one of fuzzy set or fuzzy subset, where in a fuzzy set, a membership function is defined for each element of the referential set.

Fuzzy Set: A fuzzy set $\tilde{A}$ in a universe of discourse X is defined as the set of pairs

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) : x \in X\},$$

where $\mu_{\tilde{A}} : X \to [0,1]$ is a mapping and $\mu_{\tilde{A}}(x)$ is called the membership function of $\tilde{A}$ or grade of membership of x in $\tilde{A}$.

Convex Fuzzy Set: A fuzzy set $\tilde{A}$ is called convex if and only if for all

$x_1, x_2 \in X$, $\mu_{\tilde{A}}(\lambda x_1 + (1-\lambda)x_2) \geq \min\{\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)\}$,

where $\lambda \in [0,1]$.

Support of a fuzzy set: The support of fuzzy set $\tilde{A}$ denoted by $S(\tilde{A})$ is the crisp set of all $x \in X$ such that, $\mu_{\tilde{A}}(x) > 0$.

α **-level Set:** The set of elements that belong to the fuzzy set $\tilde{A}$ at least to the degree α is called the α -level set or α -cut and is given by

$$\tilde{A}_\alpha = \{x \in X : \mu_{\tilde{A}}(x) \geq \alpha\}.$$

If $\tilde{A}_\alpha = \{x \in X : \mu_{\tilde{A}}(x) > \alpha\}$, it is called strong α -level set or strong α -cut.

Normal Fuzzy Set: A fuzzy set $\tilde{A}$ is called a normal fuzzy set if there has at least one $x \in X$ such that $\mu_{\tilde{A}}(x) = 1$.

Fuzzy Number: A fuzzy number is a fuzzy set which is both convex and normal as well as a special case of a fuzzy set. Different definitions and properties of fuzzy numbers are encountered in the literature but they all agree on that a fuzzy number represents the notion of a set of real numbers 'closer to a' where 'a' is the number being fuzzified.

Triangular Fuzzy Number (TFN)

A triangular fuzzy number $\tilde{A}$ is represented by the triplet (a_1, a_2, a_3) and defined by its continuous membership function $\mu_{\tilde{A}}(x) : X \to [0,1]$ given by

$$\mu_{\tilde{A}}(x) = \begin{cases} \dfrac{x-a_1}{a_2-a_1} & \text{if } a_1 \leq x \leq a_2 \\ 1 & \text{if } x = a_2 \\ \dfrac{a_3-x}{a_3-a_2} & \text{if } a_2 \leq x \leq a_3 \\ 0 & \text{otherwise} \end{cases}$$

Parabolic Fuzzy Number (PFN)

A parabolic fuzzy number $\tilde{A}$ is represented by the triplet (a_1, a_2, a_3) and defined by its continuous

membership function $\mu_{\tilde{A}}(x): X \rightarrow [0,1]$ given by

$$\mu_{\tilde{A}}(x)=\begin{cases}1-\left(\dfrac{a_2-x}{a_2-a_1}\right)^2 & \text{if } a_1 \le x \le a_2\\ 1 & \text{if } x=a_2\\ 1-\left(\dfrac{x-a_2}{a_3-a_2}\right)^2 & \text{if } a_2 \le x \le a_3\\ 0 & \text{otherwise}\end{cases}$$

where, $L(x)$ and $R(x)$ are continuous functions of x. Moreover, $L(x)$ is strictly monotonic increasing and $R(x)$ strictly monotonic decreasing function of x in $a_1 \le x \le a_2$ and $a_3 \le x \le a_4$ respectively.

Trapezoidal Fuzzy Number (TrFN)

A trapezoidal fuzzy number $\tilde{A}$ is represented by the quadruplet (a_1, a_2, a_3, a_4) and defined by its continuous membership function $\mu_{\tilde{A}}(x): X \rightarrow [0,1]$ given by

$$\mu_{\tilde{A}}(x)=\begin{cases}\dfrac{x-a_1}{a_2-a_1} & \text{if } a_1 \le x \le a_2\\ 1 & \text{if } a_2 \le x \le a_3\\ \dfrac{a_4-x}{a_4-a_3} & \text{if } a_3 \le x \le a_4\\ 0 & \text{otherwise}\end{cases}$$

Generalized Fuzzy Number

The generalized fuzzy number $\tilde{A}$ with membership function $\mu_{\tilde{A}}(x)$ (Fig. 1) exhibits a fuzzy subset of the real line $\mathbb{R}$, where

$$\mu_{\tilde{A}}(x)=\begin{cases}L(x) & a_1 \le x \le a_2\\ 1 & a_2 \le x \le a_3\\ R(x) & a_3 \le x \le a_4\\ 0 & \text{otherwise}\end{cases}$$

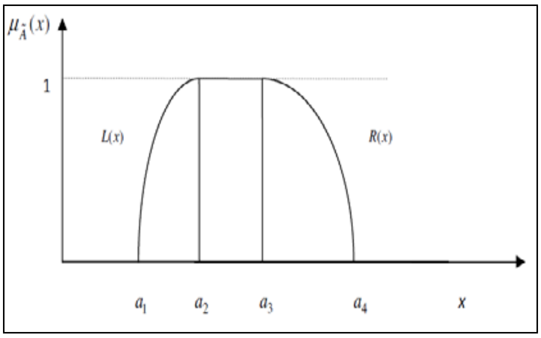

FIG. 1 GENERALIZED FUZZY NUMBER

Defuzzification

Defuzzification is the process of producing a quantifiable result in fuzzy logic, given the fuzzy sets and the corresponding degrees of membership. There are several defuzzification techniques available in the existing literature. However, the common and useful ones are as follows:

(i) Centre of Area (COA)
(ii) Bisector of Area (BOA)
(iii) Smallest of Maxima (SOM)
(iv) Largest of Maxima (LOM)
(v) Mean of Maxima (MOM)
(vi) Regular Weighted Point (RWP)
(vii) Graded Mean Integration Value (GMIV)
(viii) Centre of Approximated Interval (COAI)

Centre of Area (COA)

This is the most commonly used technique. This defuzzification can be expressed as

$$x_{COA}=\frac{\int_x x\mu_{\tilde{A}}(x)\,dx}{\int_x \mu_{\tilde{A}}(x)\,dx}$$

where x_{COA} is the crisp output, $\mu_{\tilde{A}}(x)$ is the aggregated membership function and x is the output variable. This method is also known as center of gravity or centroid defuzzification method. The only disadvantage of this method is that it is computationally difficult for complex membership functions.

Bisector of Area

The bisector of area is the vertical line that divides the region into two sub-regions of equal area. The formula for x_{BOA} is given by

$$\int_{a_1}^{x_{BOA}} \mu_{\tilde{A}}(x)\,dx = \int_{x_{BOA}}^{a_4} \mu_{\tilde{A}}(x)\,dx.$$

It is sometimes, but not always coincident with the centroid line.

Largest of Maxima

The maximum x_{LOM} takes the largest amongst all x that belong to $[a_2, a_3]$.

Smallest of Maxima

The smallest output with the maximum membership function is taken as the crisp value and it is denoted by x_{SOM}.

Mean of Maxima

In this method only active rules with the highest degree of fulfillment are taken into account. The output is computed as $x_{MOM}=\frac{1}{2}(x_{LOM}+x_{SOM})$.

Regular Weighted Point (RWP)

For the fuzzy number $A=\left(a_1,a_2,a_3,a_4\right)$, the $\alpha-$cut is $A_\alpha=\left[L_A\left(\alpha\right),R_A\left(\alpha\right)\right]$ and the regular weighted point for A is given by (Saneifard 2012)

$$RWP(A)=\frac{\int_0^1\left(\frac{L_A\left(\alpha\right)+R_A\left(\alpha\right)}{2}\right)f\left(\alpha\right)d\alpha}{\int_0^1 f\left(\alpha\right)d\alpha}$$

$$=\int_0^1\left(L_A\left(\alpha\right)+R_A\left(\alpha\right)\right)f\left(\alpha\right)d\alpha$$

where $f\left(\alpha\right)=\begin{cases}1-2\alpha & \text{when } \alpha\in\left[0,1/2\right]\\ 2\alpha-1 & \text{when } \alpha\in\left[1/2,1\right]\end{cases}$.

Graded Mean Integration Value of Fuzzy Number

For the generalized fuzzy number $\tilde{A}$ with membership function $\mu_{\tilde{A}}(x)$, according to Chen et al. (1999), the Graded Mean Integral Value $P_{dGw}(\tilde{A})$ of $\tilde{A}$ is given by

$$P_{dGw}(\tilde{A})=\frac{\int_0^1 x\left\{(1-w)L^{-1}(x)+wR^{-1}(x)\right\}dx}{\int_0^1 x dx}$$

$$=2\int_0^1 x\left\{(1-w)L^{-1}(x)+wR^{-1}(x)\right\}dx$$

where the pre-assigned parameter $w\in[0,1]$ refers to the degree of optimism. $w=1$ represents an optimistic point of view, $w=0$ represents a pessimistic point of view and $w=0.5$ indicates a moderately optimistic decision makers' point of view.

Centre of the Approximated Interval (COAI)

Let $\tilde{A}$ be a fuzzy number with interval of confidence at the level α, then the α-cut is $[A_L(\alpha),A_R(\alpha)]$. The nearest interval approximation of $\tilde{A}$ is with respect to the distance metric d is $C_d(\tilde{A})=\left[\int_0^1 A_L(\alpha)d\alpha,\int_0^1 A_R(\alpha)d\alpha\right]$, where

$$d(\tilde{A},\tilde{B})=\sqrt{\int_0^1\left\{A_L(\alpha)-B_L(\alpha)\right\}^2d\alpha+\int_0^1\left\{A_R(\alpha)-B_R(\alpha)\right\}^2d\alpha}.$$

The interval approximation for the triangular fuzzy number $\tilde{A}=(a_1,a_2,a_3)$ is $\left[\frac{(a_1+a_2)}{2},\frac{(a_2+a_3)}{2}\right]$, for the parabolic fuzzy number $\tilde{A}=(a_1,a_2,a_3)$ is $\left[\frac{1}{3}(2a_1+a_2),\frac{1}{3}(a_2+2a_3)\right]$ and for the trapezoidal fuzzy number $\tilde{A}=(a_1,a_2,a_3,a_4)$ is $\left[\frac{1}{2}(a_1+a_2),\frac{1}{2}(a_3+a_4)\right]$.

The defuzzification formulas for different fuzzy numbers are given in the Table 1. Also, the positions of defuzzifying points of different defuzzification methods are shown in Fig. 2.

TABLE 1 DEFUZZIFICATION FORMULAS FOR DIFFERENT FUZZY NUMBERS

Defuzzifier	TFN	PFN	TrFN
COA	$\frac{1}{3}(a_1+a_2+a_3)$	$\frac{1}{8}(3a_1+2a_2+3a_3)$	$\frac{1}{3}\left[a_1+a_2+a_3+a_4-\frac{a_3a_4-a_1a_2}{(a_4+a_3-a_2-a_1)}\right]$
BOA	$\frac{1}{4}(a_1+2a_2+a_3)$	$\frac{1}{3}(a_1+a_2+a_3)$	$\frac{1}{4}(a_1+a_2+a_3+a_4)$
MOM	a_2	a_2	$\frac{1}{2}(a_2+a_3)$
SOM	a_2	a_2	a_2
LOM	a_2	a_3	a_3
RWP	$\frac{1}{4}(a_1+2a_2+a_3)$	$a_2+\frac{2\left(\sqrt{2}+1\right)}{15}(a_3-2a_2+a_1)$	$\frac{1}{4}(a_4+a_3+a_2+a_1)$
GMIV (with $w=0.5$)	$\frac{1}{6}(a_1+4a_2+a_3)$	$\frac{1}{15}(4a_1+7a_2+4a_3)$	$\frac{1}{6}(a_1+2a_2+2a_3+a_4)$
COAI	$\frac{1}{4}(a_1+2a_2+a_3)$	$\frac{1}{3}(a_1+a_2+a_3)$	$\frac{1}{4}(a_1+a_2+a_3+a_4)$

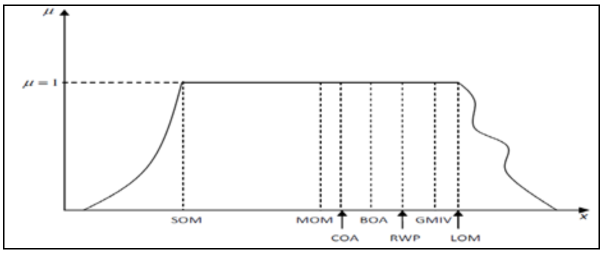

FIG. 2 POSITION OF DEFUZZIFYING POINTS IN DIFFERENT DEFUZZIFICATION METHODS

Mathematical Formulation of the Problem

Let us consider a redundancy allocation problem of complicated reliability system with ten-component. Each subsystem is connected parallelly with identical components. Our objective is to maximize the overall system reliability subject to the given resource constraints. This can be done by finding the number of redundant components in each subsystem.

The general form of the redundancy allocation problem in crisp form is as follows:

$$\text{Maximize } R_S(x) \tag{1}$$

subjected to $g_i(x)\le b_i$, $i=1,2,\cdots,m$, where $x=(x_1,x_2,...,x_n)$, $1\le l_j\le x_j\le u_j$, x_j is integer, $j=1,...,n$, and b_i is the i-th available resource, $i=1,2,\cdots,m$.

Now, if the component reliabilities are imprecise, then the reliability of each subsystem and finally the overall system reliability will be imprecise. In this situation, the general form of the redundancy allocation problem can be written as follows:

$$\text{Maximize } \tilde{R}_S(x) \tag{2}$$

subjected to $\tilde{g}_i(x)\le\tilde{b}_i$, $i=1,2,\cdots,m$ where $x=(x_1,x_2,...,x_n)$, $1\le l_j\le x_j\le u_j$, x_j is integer, $j=1,...,n$ and $\tilde{b}_i$ is the i-th available resource which is imprecise ,

$i = 1, 2, \cdots, m.$

Fuzzy number representation of reliability of each component as well as different parameters of resource constraints is used to characterize the impreciseness of the system.

Genetic Algorithm Based Constraints Handling Approach

Evidently the problem (2) is a constrained optimization problem. Several techniques have been projected to deal with the constraints for solving the optimization problem. Recently, Gupta et al. (2009) and Bhunia et al. (2010) solved the optimization problem using Big-M penalty technique. In this method, the given constrained optimization problem is converted to an unconstrained one by penalizing a large positive number say, M . This method is known as Big-M penalty technique. In this work, the Big-M penalty technique has been used to handle the constraints.

The corresponding unconstrained optimization problem is as follows:

$$\text{Maximize } defuz\left(\tilde{R}_S\right) = \begin{cases} defuz\left(\tilde{R}_S\right) & \text{if } x \in S \\ -M & \text{if } x \notin S \end{cases} \quad (3)$$

where

$$S = \left\{ x : defuz\left(\tilde{g}_j(x_1, x_2, ..., x_n)\right) \le defuz\left(\tilde{b}_j\right),\ j = 1, 2, ..., m \right\}$$

be the feasible space. Here, $defuz\left(\tilde{A}\right)$ denotes the defuzzified value of the fuzzy number $\tilde{A}$. The problem (3) is a non-linear unconstrained optimization problem with defuzzified objective of n integer variables $x_1, x_2, ..., x_n$. In the next section, a solution procedure is prepared to solve the problem (3).

Solution Procedure

Since the problem in (3) is non-linear in nature with integer variables, it is difficult to reach an analytic solution (if any) to the problem (Gen and Cheng 1997). Furthermore, efficient treatment of integer non-linear optimization problem is one of the most difficult problems in practical optimization (El-Sharkawi 2008). Hence, to solve it, a meta-heuristic algorithm should be employed.

A number of researchers have successfully used meta-heuristic algorithms to solve the complicated optimization problems in different fields of scientific and engineering disciplines. Some of these meta-heuristic algorithms are tabu search, genetic algorithm, particle swarm optimization, ant colony optimization and bee colony optimization among which genetic algorithm is the most popular one.

Genetic Algorithm (GA)

Genetic Algorithm (GA) is a stochastic search and optimization technique (Goldberg 1989). Gen and Cheng (1996) described the application of GA to combinatorial problems including reliability optimization problems. Perhaps, GA is the most widely known evolutionary computation method due to its simplicity, powerfulness and wide application. It works by the evolutionary principles and chromosomal processing in natural genetics. GA maintains a population $P(t)$ for generation t, of chromosomes which are the set of genes, the part of solution.

The most fundamental idea of Genetic Algorithm is to replicate the natural evolution process artificially in which populations undergo continuous changes through genetic operators, like crossover, mutation and selection. In particular, it is very useful to solve complicated optimization problems which cannot be solved easily by direct or gradient based mathematical techniques. It is very effective to handle large-scale, real-life, discrete and continuous optimization problems without making unrealistic assumptions and approximations.

The approach of GA is applied to many engineering problem as well as decision making problems in various fields. Some distinguished characteristics of GA are as follows (Goldberg 1989):

(i) GA works with a coding of solution set, not the solution themselves,

(ii) GA searches over a population of solutions, not a single solution,

(iii) GA uses payoff information, not derivatives or other auxiliary knowledge,

(iv) GA applies stochastic transformation rules, but not deterministic.

Each chromosome represents a potential solution to the problem under consideration and is evaluated to give some measure of its fitness. Some chromosomes undergo stochastic transformation by means of genetic operators to form new chromosomes. The genetic operator 'crossover' creates new chromosomes by combining some parts from two chromosomes and the other genetic operator 'mutation' produces offspring by making changes in a single chromosome. A new population is formed by selecting the chromosomes having better fitness value from the parent and the offspring populations taken together. After several

generations, the algorithm converges to the best chromosome having the best fitness value and it represents the optimal solution.

The algorithm to implement GA is as follows:

Step-1: Set population size (*popsize*), maximum number of generations (*maxgen*), probability of crossover (*pcros*), probability of mutation (*pmute*) and the bounds of decision variables.

Step-2: Set *t=0*.

Step-3: Initialize the chromosomes of the population $P(t)$.

Step-4: Compute the fitness function (objective function) for each chromosome of $P(t)$.

Step-5: Find the chromosome having the best fitness value.

Step-6: Set *t=t +1*.

Step-7: If the termination condition is satisfied, then go to Step-13; otherwise the next step.

Step-8: Select the population $P(t)$ from the population $P(t-1)$ of *(t-1)*- th generation using tournament selection.

Step-9: Apply crossover & mutation operators on $P(t)$ to produce new population $P(t)$.

Step-10: Compute the fitness function value of each chromosome of $P(t)$.

Step-11: Find the best chromosome from $P(t)$.

Step-12: Find the better of the best chromosomes of $P(t)$ & $P(t-1)$ and store it; go to Step-6.

Step-13: Print the best chromosome and its fitness value

Step-14: End.

The following vital components have been taken into account in the application of genetic algorithm.

(i) GA parameters (population size, maximum number of generation, crossover rate and mutation rate)

(ii) Chromosome representation

(iii) Initialization of population

(iv) Evaluation of fitness function

(v) Selection process

(vi) Genetic operators (crossover and mutation)

GA Parameters

In terms of implementing GA, the following GA parameters has been taken into account:

Population size (popsize): Population size determines the amount of information stored by the GA. There is no clear rule how large it should be. The population size is problem dependent and will need to increase/ decrease with the dimension of the problem.

Maximum number of generations (maxgen): It varies from problem to problem and depends upon the number of genes (variables) of a chromosome and it is prescribed to be the termination criterion for convergence of the solution.

Probability of crossover (pcross): In Genetic Algorithms crossover is considered to be the main search operator. The crossover operator is used to thoroughly explore the search process. In crossover operator the genetic information between/among two or more individuals are blended to produce new individuals. Normal range of the crossover rate lies in [0.60, 0.95].

Probability of mutation (pmute): Mutation operator plays an important role in genetic algorithm. After crossover operation, mutation is performed. It is intended to prevent to the falling of all solutions in the population into a local optimum of the problem under consideration. Mutation operator randomly changes the offspring resulted from crossover. The mutation rate lies in [0.05, 0.20]. Sometimes mutation rate is considered to be $1/n$ where n is the number of genes (variables) of the chromosome.

Chromosome Representation

In the existing literature there are different types of representations of chromosomes, like, binary, real, octal, hexadecimal coding, amongst which real coding representations are well accepted. In this representation, a chromosome is coded in the form of vector/matrix of integer/floating point or combination of the both numbers and every component of that chromosome represents a decision variable of the problem. In this representation, each chromosome is encoded as a vector of integer numbers, with the same component as the vector of decision variables of the problem under consideration. This type of representation is accurate and more efficient as it is closed to the real design space and moreover, the string length of each chromosome is same as the number of decision variables. In this representation, for a given problem with n decision variables, a n-component vector $x=(x_1,x_2,\cdots,x_n)$ is used as a chromosome to represent a solution to the problem. A chromosome denoted as v_k ($k=1,2,...,p\_size$) is an ordered list of n genes as $v_k=\{v_{k1},v_{k2},...,v_{ki},...,v_{kn}\}$.

Initialization of Population

After representation of chromosome, the next step is to initialize the chromosome that will take part in the artificial genetics. To initialize the population, first of all we have to find the independent variables and their bounds for the given problem. Then the initialization process produces population size number of chromosomes in which every component for each chromosome is randomly generated within the bounds of the corresponding decision variable. There are several procedures to select a random number of integer type. In this work, the following algorithm has been applied to select of an integer random number.

An integer random number between a and b can be generated as either $x = a + g$ or, $x = b - g$, where g is a random integer between 1 and $|a-b|$.

Evaluation of Fitness Function

Evaluation of fitness function is same for natural evolution process in the biological and physical environments. After initialization of chromosomes of potential solutions, we need to see how relatively good they are. Therefore, the fitness value for each chromosome has to be calculated. In our work, the value of objective function of the reduced unconstrained optimization problems corresponding to the chromosome is considered as the fitness value of that chromosome.

Selection of Fitness Function

The selection operator which is the first operator in artificial genetics plays an interesting role in GA. This selection process is based on the Darwin's principle on natural evolution "survival of the fittest". The primary objective of this process is to select the above average individuals/chromosomes from the population according to the fitness value of each chromosome and eliminate the rest of the individuals/chromosomes. There are several methods to implement the selection process.

In this work, the well known tournament selection with size two has been utilized.

Genetic Operators

After the selection process, other genetic operators, like crossover and mutation are applied to the resulting chromosomes those which have survived. Crossover is an operator that creates new individuals/ chromosomes (offspring) by combining the features of both parent solutions. It operates on two or more parent solutions at a time and produces offspring for next generation. In this work, intermediate crossover for integer variables has been used.

The aim of mutation operator is to introduce the random variations into the population used to prevent the search process from converging to the local optima. This operator helps to regain the information lost in earlier generations and is responsible for fine tuning capabilities of the system and is applied to a single individual only. Usually, its rate is very low; because otherwise it would defeat the order building generated through the selection and crossover operations. In this work one-neighborhood mutation has been in use.

Numerical Example

The mathematical formulation of the redundancy allocation problem of a complex reliability system with ten subsystems is as follows

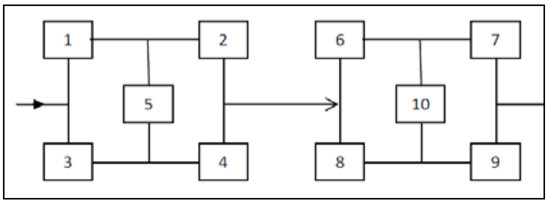

FIG. 3 DIAGRAM OF A COMPLEX SYSTEM

$$\text{Max } \tilde{R}_s = \tilde{R}1_s(x)\tilde{R}2_s(x)$$

subjected to

$$\sum_{j=1}^{10} \tilde{p}_j x_j^2 \le \tilde{P}, \ \sum_{j=1}^{10} \tilde{c}_j \left[x_j + \exp\left(\frac{x_j}{4}\right)\right] \le \tilde{C},$$

$$\sum_{j=1}^{10} \tilde{w}_j x_j \exp\left(\frac{x_j}{4}\right) \le \tilde{W} \text{ and } x_j \ge 1, \ j = 1, 2, \ldots, 10$$

where

$$\tilde{R}1_S(x) = \tilde{R}_1\tilde{R}_2 + \tilde{Q}_2\tilde{R}_3\tilde{R}_4 + \tilde{Q}_1\tilde{R}_2\tilde{R}_3\tilde{R}_4 + \tilde{R}_1\tilde{Q}_2\tilde{Q}_3\tilde{R}_4\tilde{R}_5 + \tilde{Q}_1\tilde{R}_2\tilde{R}_3\tilde{Q}_4\tilde{R}_5$$

and

$$\tilde{R}2_S(x) = \tilde{R}_6\tilde{R}_7 + \tilde{Q}_7\tilde{R}_8\tilde{R}_9 + \tilde{Q}_6\tilde{R}_7\tilde{R}_8\tilde{R}_9 + \tilde{R}_6\tilde{Q}_7\tilde{Q}_8\tilde{R}_9\tilde{R}_{10} + \tilde{Q}_6\tilde{R}_7\tilde{R}_8\tilde{Q}_9\tilde{R}_{10}$$

TABLE 2 INPUT PARAMETERS VALUES FOR TFN AND PFN CASE

j	$\tilde{r}_j$	$\tilde{p}_j$	$\tilde{c}_j$	$\tilde{w}_j$
1	(0.80, 0.90, 0.95)	(4.0, 4.4, 5.0)	(6.9, 7.8, 8.5)	(7.5, 8.6, 9.2)
2	(0.60, 0.70, 0.80)	(4.5, 5.0, 5.6)	(5.4, 6.3, 7.1)	(5.3, 6.1, 7.6)
3	(0.85, 0.90, 0.96)	(3.0, 3.2, 4.0)	(7.5, 8.1, 9.0)	(7.2, 8.4, 9.3)
4	(0.65, 0.70, 0.80)	(2.6, 3.1, 3.8)	(8.0, 9.2, 9.7)	(6.4, 7.5, 8.7)
5	(0.62, 0.70, 0.79)	(3.8, 4.4, 5.3)	(6.6, 7.3, 8.3)	(5.4, 6.7, 7.8)
6	(0.75, 0.90, 0.95)	(2.1, 2.9, 3.4)	(5.7, 6.2, 7.7)	(4.3, 5.0, 6.2)
7	(0.50, 0.70, 0.79)	(2.4, 2.9, 3.7)	(5.6, 6.5, 7.3)	(4.3, 5.1, 6.4)
8	(0.76, 0.80, 0.87)	(2.6, 3.1, 4.3)	(7.3, 8.1, 8.7)	(5.5, 6.8, 7.4)
9	(0.75, 0.80, 0.86)	(1.8, 2.5, 3.5)	(8.5, 9.2, 10.0)	(4.3, 5.4, 6.7)
10	(0.70, 0.80, 0.95)	(1.7, 2.8, 3.4)	(6.0, 7.9, 8.5)	(7.6, 8.9, 9.3)
$\tilde{P}$ = (297, 300,305), $\tilde{C}$ = (396, 400, 410), $\tilde{W}$ =(398, 400, 404)				

TABLE 3 INPUT PARAMETERS VALUES FOR TrFN CASE

j	$\tilde{r}_j$	$\tilde{p}_j$	$\tilde{c}_j$	$\tilde{w}_j$
1	(0.80, 0.90, 0.95, 0.98)	(4.0, 4.4, 5.0, 5.5)	(6.9, 7.8, 8.5, 9.0)	(7.5, 8.6, 9.2, 9.6)
2	(0.60, 0.70, 0.80, 0.85)	(4.5, 5.0, 5.6, 6.0)	(5.4, 6.3, 7.1, 7.5)	(5.3, 6.1, 7.6, 8.3)
3	(0.85, 0.90, 0.96, 0.86)	(3.0, 3.2, 4.0, 4.5)	(7.5, 8.1, 9.0, 9.2)	(7.2, 8.4, 9.3, 9.7)
4	(0.65, 0.70, 0.80, 0.86)	(2.6, 3.1, 3.8, 4.3)	(8.0, 9.2, 9.7, 9.8)	(6.4, 7.5, 8.7, 9.1)
5	(0.62, 0.70, 0.79, 0.82)	(3.8, 4.4, 5.3, 6.0)	(6.6, 7.3, 8.3, 9.0)	(5.4, 6.7, 7.8, 8.6)
6	(0.75, 0.90, 0.95, 0.97)	(2.1, 2.9, 3.4, 4.0)	(5.7, 6.2, 7.7, 8.3)	(4.3, 5.0, 6.2, 7.4)
7	(0.50, 0.70, 0.79, 0.83)	(2.4, 2.9, 3.7, 4.2)	(5.6, 6.5, 7.3, 8.0)	(4.3, 5.1, 6.4, 7.5)
8	(0.76, 0.80, 0.87, 0.91)	(2.6, 3.1, 4.3, 4.6)	(7.3, 8.1, 8.7, 8.9)	(5.5, 6.8, 7.4, 8.3)
9	(0.75, 0.80, 0.86, 0.90)	(1.8, 2.5, 3.5, 3.8)	(8.5, 9.2, 10.0, 10.2)	(4.3, 5.4, 6.7, 7.2)
10	(0.70, 0.80, 0.95, 0.97)	(1.7, 2.8, 3.4, 4.0)	(6.0, 7.9, 8.5, 9.0)	(7.6, 8.9, 9.3, 9.8)
$\tilde{P}$ = (297, 300,305,308), $\tilde{C}$ = (396, 400, 410,413), $\tilde{W}$=(398, 400, 404,407)				

In all the cases, the problem has been fixed by real coded advanced genetic algorithm. In this algorithm, we have used tournament selection, intermediate crossover and one neighborhood mutation as genetic operators. For this purpose, the code has been prepared for this algorithm in C Programming language. The corresponding computational work has been done on a PC with Intel Core-2 duo processor in LINUX environment. For each problem, 20 independent runs have been performed to determine the best found system reliability which is nothing but the optimal value of system reliability. In this computation, the values of genetic parameters, like *popsize*, *maxgen*, *pcross* and *pmute* have been taken as 300, 300, 0.85 and 0.15 respectively.

The computational results have been shown in Tables 4-6 for different parametric values. From Table 4, it follows that in the case of TFN parametric values, the best optimum system reliability is obtained in the cases of MOM, SOM and LOM. In case of PFN parametric values, the best optimum system reliability is obtained in cases of MOM, SOM and LOM as in the case of TFN parametric values and given in Table 5. Table 6 represents the results in the case of TrFN parametric values and indicates that the best system reliability is available in the case LOM. All the results have been shown in Fig. 4.

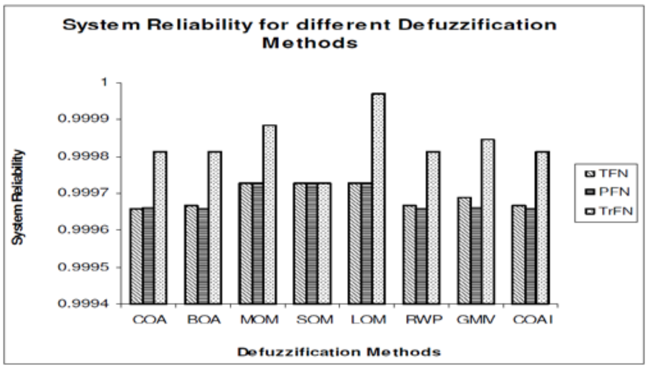

FIG. 4 SYSTEM RELIABILITY FOR DIFFERENT DEFUZZIFICATION METHODS

TABLE 4 RESULTS FOR TFN CASE

Defuzzification method	$X=(x_i, i=1,2,\ldots,10)$	System reliability (R)	Computational time (sec.)
COA	2334133341	0.99965957	0.79
BOA	2424133341	0.99966688	0.79
MOM	2434134231	0.99972740	0.79
SOM	2434134231	0.99972740	0.79
LOM	2434134231	0.99972740	0.79
RWP	2424133341	0.99966688	0.79
GMIV	2434124331	0.99968772	0.79
COAI	2424133341	0.99966688	0.79

TABLE 5 RESULTS FOR PFN CASE

Defuzzification method	$X=(x_i, i=1,2,\ldots,10)$	System reliability (R)	Computational time (sec.)
COA	2334133341	0.99966204	0.79
BOA	2334133341	0.99965957	0.79
MOM	2434134231	0.99972740	0.79
SOM	2434134231	0.99972740	0.79
LOM	2434134231	0.99972740	0.79
RWP	2334133341	0.99965938	0.79
GMIV	2424133341	0.99966181	0.79
COAI	2334133341	0.99965957	0.79

TABLE 6 RESULTS FOR TrFN CASE

Defuzzification method	$X=(x_i, i=1,2,\ldots,10)$	System reliability (R)	Computational time (sec.)
COA	2334123341	0.99981271	0.79
BOA	2424134231	0.99981413	0.79
MOM	2334134231	0.99988335	0.79
SOM	3424134231	0.99972740	0.79
LOM	2324134231	0.99997059	0.79
RWP	2424134231	0.99981413	0.79
GMIV	2334123341	0.99984581	0.79
COAI	2424134231	0.99981413	0.79

Concluding Remarks

The reliability of a component in a system might be imprecise rather than precise for several reasons. This impreciseness may be represented by diverse ways. In this paper, we have represented this by fuzzy number. Then the problem has been converted into crisp nonlinear programming problem after defuzzification in which the objective functions and all the constraints are crisp valued. For further research, one may apply other heuristic methods, like, DE, SA, PSO, etc. to solve the problem discussed in this paper. Also, the proposed technique may be applied in solving the realistic engineering and other optimization problems.

ACKNOWLEDGEMENT

The first author acknowledges the financial support provided by the University Grants Commission, India (vide sanction order No. PSW-006/12-13(ERO) dated 05.02.2013) to carry out this research work.

REFERENCES

Aggarwal, M., Gupta, R. “Penalty function approach in heuristic algorithms for constrained redundancy reliability optimization”, IEEE Transactions on Reliability, 54(3), pp.549-558, 2005.

Bellmann, R.E., Zadeh, L.A. “Decision making in a fuzzy environment”, Management Science, 17(4), B141-B164,1970.

Bhunia, A.K., Sahoo, L., and Roy, D. “Reliability stochastic optimization for a series system with interval component reliability via genetic algorithm”, Applied Mathematics and Computations, 216, pp. 929-939, 2010.

Chen, D. X. “Fuzzy reliability of the bridge circuit system”, System Engineering-Theory and Practice, 11, pp. 109-112, 1997.

Chen, S. H. “Operation on fuzzy numbers with function principal”, Tamkang Journal of Management Science, 6(1), pp. 13-25, 1985.

Chen, S. H., Hsieh, C. H. “Graded mean integration representation of generalized fuzzy numbers”, Journal of Chinese Fuzzy Systems Association, 5(2), pp.1-7, 1999.

Chen, S. M. “ Fuzzy group decision making for evaluating the rate of aggregative risk in software development”, Fuzzy Sets and Systems, Elsevier, (118), pp. 75-88,2001.

Chen, T. “IAs based approach for reliability redundancy allocation problems” Applied Mathematics and Computation, 182, pp.1556-1567, 2006.

Chern, M.S. “On the computational complexity of reliability redundancy allocation in a series system”, Operations Research Letter, 11, pp.309-315, 1992.

El-Sharkawi, L. “Modern heuristic optimization techniques (1st ed.)”, Wiley Inter Science, New Jersey, USA, 2008.

Gen, M., and Cheng, R. “Genetic algorithm and engineering design (1st ed.)”, USA, John Wiley & Sons, New York, NY, 1997.

Goldberg, D.E. “ Genetic Algorithms: Search, Optimization and Machine Learning, Reading, MA”, Addison Wesley, 1989.

Gopal, K., Aggarwal, K.K. and Gupta, J.S. “An improved algorithm for reliability optimization”, IEEE Transactions on Reliability, 29, pp. 325-328, 1978.

Gopal, K., Aggarwal, K.K. and Gupta, J.S. “A new method for reliability optimization problem”, IEEE Transactions on Reliability, 29, pp. 36-37, 1980.

Grzegorzewski, P. “Nearest interval approximation of a fuzzy number”, Fuzzy Sets and Systems, 1320, pp. 321-330, 2002.

Gupta, R.K, Bhunia, A.K. and Roy, D. “A GA Based penalty function technique for solving constrained Redundancy allocation problem of series system with interval valued reliabilities of components”, Journal of Computational and Applied Mathematics, 232, pp. 275-284,2009.

Hikita, M., Nakagawa, K., Nakashima, K. and Narihisa, H. “Reliability Optimization of Systems by a Surrogate-Constraints Algorithm”, IEEE Transactions on Reliability, 41(3),pp. 473-480, 1992.

Huang, H. Z. “Fuzzy multi-objective optimization decision-making of reliability of series system”, Microelectronics Reliability, 37(3), pp. 447-449, 1997.

Hwang, C. L., Tillman, F.A. and Kuo, W. “Reliability Optimization by Generalized lagrangian-function based and reduced-gradient methods”, IEEE Transactions on Reliability, 28, pp. 316-319, 1989.

Kim, J. H., & Yum, B. J. “A Heuristic Method for Solving Redundancy Optimization Problems in Complex Systems”, IEEE Transactions on Reliability, 42(4), pp. 572-578, 1993.

Kuo, W. and Prasad, V.R. “An annoted overview of system-reliability optimization”, IEEE Transactions on Reliability, 49,pp. 176-187, 2001.

Kuo, W., Prasad, V.R., Tillman, F.A. and Hwang, C. L. “Optimal reliability design: Fundamentals and Application”, Cambridge, Cambridge University Press, 2001.

Liou, T. S. and Wang, M. J. J. “Ranking fuzzy numbers with integral value”, Fuzzy sets and Systems, 50, pp. 247-255, 1992.

Mahapatra, G.S. and Roy, T.K. “Reliability evaluation of bridge system with fuzzy reliability of components using interval nonlinear programming”, Electronic Journal of Applied Statistical Analysis, 5(2), pp. 151-163, 2012.

Michalewicz, Z. “Genetic Algorithms +Data Structure = evolution Programs”, Springer-Verlag, Berlin, 1996.

Miettinen, K., Makela, M.M. and Toivanen, J. “Numerical comparison of some Penalty-Based Constraint Handling Techniques in Genetic Algorithms”, Journal of Global Optimization, 27, pp. 427-446, 2003.

Ming, M., Kandel, A. and Friedman, M. “A new approach for defuzzification”, Fuzzy sets and Systems, 111, 351-356, 2000.

Misra, K.B. and Sharma, U. “An efficient algorithm to solve integer- programming problems arising in system-reliability design”, IEEE Transactions on Reliability, 40,

pp. 81-91, 1991.

Nakagawa, Y. and Miyazaki, S. "Surrogate Constraints Algorithm for Reliability Optimization Problems with Two Constraints", IEEE Transactions on Reliability, 30, pp. 175-180, 1981.

Nakagawa, Y. and Nakashima, K. "A Heuristic Method for Determining Optimal Reliability Allocation", IEEE Transactions on Reliability, 26(3), pp. 156-161, 1977.

Nazz, S., Alam, A. and Biswas, R. "Effect of different defuzzification methods in a fuzzy based load balancing application", International Journal of Computer Science, 8(5), pp. 261-267, 2011.

Park, S. "Fuzzy apportionment of system reliability", IEEE Transactions on Reliability, 36, pp.129-132, 1987.

Ravi, V., Murty, N.B.S., & Reddy, P. J. "Non-equilibrium simulated annealing-algorithm applied to reliability optimization of complex systems", IEEE Transactions on Reliability, 46, pp. 233-239, 1997.

Sahoo, L., Bhunia, A. K. and Kapur, P. K. "Genetic algorithm based multi-objective reliability optimization in interval environment", Computers & Industrial Engineering, 62, pp. 152-160, 2012.

Sahoo, L., Bhunia, A.K. and Roy, D. "A Genetic Algorithm Based Reliability Redundancy Optimization for Interval valued Reliabilities of components", Journal of Applied Quantitative Methods, 5(2), pp.270-287, 2010.

Saneifard, R. "Another method for defuzzification based on regular weighted point", International Journal of Industrial Mathematics, 4(2), pp.147-152, 2012.

Saneifard, R. "A method for defuzzification by weighted distance", International Journal of Industrial Mathematics, 3, pp. 209-217, 2009.

Shelokar, P.S., Jayaraman, V. K. and Kulkarni, B. D. "Ant Algorithm for single and multiobjective reliability optimization problems", Quality and Reliability Engineering International, 18, pp. 497-514, 2002.

Sun, X. L. and Li, D. "Optimization condition and branch and bound algorithm for constrained redundancy optimization in series system", Optimization and Engineering, 3, pp.53-65, 2002.

Sun, X. L., Mckinnon, K.I.M. and Li, D. "A convexification method for a class of global optimization problems with application to reliability optimization", Journal of Global Optimization, 21, pp. 185-199, 2001.

Tillman, F. A., Hwang, C.L. and Kuo, W. "Optimization technique for system reliability with redundancy: A Review", IEEE Transactions on Reliability, 26, pp. 148-155, 1977a.

Tillman, F. A., Hwang, C.L. and Kuo, W. "Determining component reliability and redundancy for optimum system reliability", IEEE Transactions on Reliability, 26(3), pp. 162-165, 1977b.

Yager, R. R. and Filev, D. P. "On the issue of defuzzification and selection based on a fuzzy set", Fuzzy sets and Systems, 55, pp. 255-272, 1993.

Zadeh, L. A. "Fuzzy Sets", Information and Control, 8, pp. 338-353, 1965.

Zimmermann, H. J. "Description and Optimization of Fuzzy Systems", International Journal of General Systems, 2, pp. 209-215, 1976.

Zimmermann, H. J. "Fuzzy sets theory and its applications", Kluwer Academic Press, Dordrecht, 1991.

Sanat Kumar Mahato is an Assistant Professor of Mathematics Durgapur Govt. College, West Bengal, India. His area of research work includes Application of Genetic Algorithm in reliability optimization.

Laxminarayan Sahoo is an Assistant Professor in the Department of Mathematics, Raniganj Girls' College, India. He obtained his M.Sc. degree in Applied Mathematics from Vidyasagar University, India and obtained the Ph.D. in Mathematics from The University of Burdwan, India. His research interest includes reliability optimization, interval mathematics, interval ranking, fuzzy mathematics and applications of Genetic Algorithm and Particle Swarm Optimization in the areas of decision-making problems. He has also received Prof. M. N. Gopalan Award for best Ph.D. thesis in Operations Research from Operational Research Society of India (ORSI)

A.K. Bhunia is an Associate Professor of Mathematics in the Department of Mathematics, The University of Burdwan, West Bengal, India. He obtained his Ph.D. degree in Mathematics from Vidyasagar University, West Bengal, India. His fields of research are Genetic Algorithm and applications, Interval Analysis and applications, Inventory control Theory. He has published more than eighty research papers in referred national and international journals. He has also reviewed a large number of research papers of Elsevier Science, Springer-Verlag. Taylor & Francis and ASME, France published journals.